

CPlusProject

0.0.0

Generated by Doxygen 1.12.0



<b>1 CPlusProject</b>	<b>1</b>
<b>2 Hierarchical Index</b>	<b>3</b>
2.1 Class Hierarchy	3
<b>3 Class Index</b>	<b>5</b>
3.1 Class List	5
<b>4 File Index</b>	<b>7</b>
4.1 File List	7
<b>5 Class Documentation</b>	<b>9</b>
5.1 AAInteractableActor Class Reference	9
5.1.1 Constructor & Destructor Documentation	9
5.1.1.1 AAInteractableActor()	9
5.1.2 Member Function Documentation	10
5.1.2.1 BeginFocus2D()	10
5.1.2.2 EndFocus2D()	10
5.1.2.3 GetItemData()	10
5.1.3 Member Data Documentation	10
5.1.3.1 itemData	10
5.2 AFlickeringLight Class Reference	10
5.2.1 Constructor & Destructor Documentation	11
5.2.1.1 AFlickeringLight()	11
5.2.2 Member Function Documentation	11
5.2.2.1 BeginPlay()	11
5.2.2.2 Tick()	11
5.2.3 Member Data Documentation	11
5.2.3.1 spotlight	11
5.3 AGameOverUIController Class Reference	12
5.3.1 Member Function Documentation	12
5.3.1.1 BeginPlay()	12
5.3.2 Member Data Documentation	13
5.3.2.1 gameOverUIClass	13
5.4 AGameUIContoller Class Reference	13
5.4.1 Member Function Documentation	14
5.4.1.1 BeginPlay()	14
5.4.1.2 CloseAll()	14
5.4.1.3 CloseEncourageInteractUI()	14
5.4.1.4 CloseLockUI()	14
5.4.1.5 OpenEncourageInteractUI()	14
5.4.1.6 SetLockController()	14
5.4.1.7 UpdateInteractionUI()	15
5.4.2 Member Data Documentation	15

5.4.2.1 <a href="#">enourageInteractionUIClass</a>	15
5.4.2.2 <a href="#">interactionUIClass</a>	15
5.4.2.3 <a href="#">lockUIClass</a>	15
5.5 <a href="#">AMainMenuUIController Class Reference</a>	15
5.5.1 <a href="#">Member Function Documentation</a>	16
5.5.1.1 <a href="#">BeginPlay()</a>	16
5.5.2 <a href="#">Member Data Documentation</a>	16
5.5.2.1 <a href="#">mainMenuClass</a>	16
5.6 <a href="#">AMyPlayerController Class Reference</a>	16
5.6.1 <a href="#">Constructor &amp; Destructor Documentation</a>	17
5.6.1.1 <a href="#">AMyPlayerController()</a>	17
5.6.2 <a href="#">Member Function Documentation</a>	17
5.6.2.1 <a href="#">OnPossess()</a>	17
5.6.2.2 <a href="#">OnUnPossess()</a>	17
5.6.2.3 <a href="#">SetLastInteractedActor()</a>	17
5.6.3 <a href="#">Member Data Documentation</a>	18
5.6.3.1 <a href="#">actionInteract</a>	18
5.6.3.2 <a href="#">actionMove</a>	18
5.6.3.3 <a href="#">direction</a>	18
5.6.3.4 <a href="#">inputMappingContext</a>	18
5.7 <a href="#">AUIController Class Reference</a>	18
5.7.1 <a href="#">Member Function Documentation</a>	19
5.7.1.1 <a href="#">CloseAll()</a>	19
5.7.1.2 <a href="#">SwitchToPlayerControls()</a>	19
5.7.1.3 <a href="#">SwitchToUIControls()</a>	19
5.7.2 <a href="#">Member Data Documentation</a>	19
5.7.2.1 <a href="#">gameInputMode</a>	19
5.7.2.2 <a href="#">inputModeUI</a>	19
5.8 <a href="#">CplusProject Class Reference</a>	19
5.8.1 <a href="#">Constructor &amp; Destructor Documentation</a>	20
5.8.1.1 <a href="#">CplusProject()</a>	20
5.9 <a href="#">FItemData Struct Reference</a>	20
5.9.1 <a href="#">Detailed Description</a>	20
5.9.2 <a href="#">Constructor &amp; Destructor Documentation</a>	20
5.9.2.1 <a href="#">FItemData()</a>	20
5.9.3 <a href="#">Member Function Documentation</a>	21
5.9.3.1 <a href="#">GENERATED_BODY()</a>	21
5.9.4 <a href="#">Member Data Documentation</a>	21
5.9.4.1 <a href="#">description</a>	21
5.9.4.2 <a href="#">interactable</a>	21
5.9.4.3 <a href="#">itemType</a>	21
5.10 <a href="#">FSimonData Struct Reference</a>	21

5.10.1 Constructor & Destructor Documentation	22
5.10.1.1 FSimonData()	22
5.10.2 Member Function Documentation	22
5.10.2.1 GENERATED_BODY()	22
5.10.3 Member Data Documentation	22
5.10.3.1 mapColorToNote	22
5.10.3.2 sequence	22
5.11 IInteractionInterface Class Reference	22
5.11.1 Detailed Description	23
5.11.2 Member Function Documentation	23
5.11.2.1 BeginFocus2D()	23
5.11.2.2 EndFocus2D()	23
5.11.2.3 GetItemData()	23
5.12 UGameOverUI Class Reference	23
5.13 UInteractionInterface Class Reference	24
5.13.1 Detailed Description	24
5.14 UInteractionUI Class Reference	25
5.14.1 Member Function Documentation	25
5.14.1.1 UpdateWidget()	25
5.15 ULockUI Class Reference	25
5.15.1 Member Function Documentation	26
5.15.1.1 BindButtons()	26
5.15.1.2 BlueButton()	27
5.15.1.3 CheckSequence()	27
5.15.1.4 GiveHint()	27
5.15.1.5 GreenButton()	27
5.15.1.6 NativeConstruct()	27
5.15.1.7 OnClose()	27
5.15.1.8 OnOpen()	27
5.15.1.9 OnUnlocked()	27
5.15.1.10 PlaySequence()	28
5.15.1.11 RedButton()	28
5.15.1.12 SetLockController()	28
5.15.1.13 YellowButton()	28
5.15.2 Member Data Documentation	28
5.15.2.1 blueButton	28
5.15.2.2 blueHighlightImage	28
5.15.2.3 closeButton	28
5.15.2.4 errorImage	28
5.15.2.5 errorSound	28
5.15.2.6 greenButton	28
5.15.2.7 greenHighlightImage	29

5.15.2.8	<a href="#">hintButton</a>	29
5.15.2.9	<a href="#">notesAudio</a>	29
5.15.2.10	<a href="#">openDoorActorClass</a>	29
5.15.2.11	<a href="#">redButton</a>	29
5.15.2.12	<a href="#">redHighlightImage</a>	29
5.15.2.13	<a href="#">showButtonHighlightTime</a>	29
5.15.2.14	<a href="#">showErrorTime</a>	29
5.15.2.15	<a href="#">timeBetweenNotes</a>	29
5.15.2.16	<a href="#">volumeMultiplier</a>	29
5.15.2.17	<a href="#">yellowButton</a>	30
5.15.2.18	<a href="#">yellowHighlightImage</a>	30
5.16	<a href="#">UMainMenuUI Class Reference</a>	30
5.16.1	<a href="#">Member Function Documentation</a>	30
5.16.1.1	<a href="#">NativeConstruct()</a>	30
5.16.1.2	<a href="#">StartGame()</a>	31
5.16.2	<a href="#">Member Data Documentation</a>	31
5.16.2.1	<a href="#">startGameButton</a>	31
<b>6</b>	<b><a href="#">File Documentation</a></b>	<b>33</b>
6.1	<a href="#">README.txt File Reference</a>	33
6.2	<a href="#">Source/CplusplusProject/CplusplusProject.Build.cs File Reference</a>	33
6.3	<a href="#">Source/CplusplusProject/CplusplusProject.cpp File Reference</a>	33
6.3.1	<a href="#">Function Documentation</a>	33
6.3.1.1	<a href="#">IMPLEMENT_PRIMARY_GAME_MODULE()</a>	33
6.4	<a href="#">Source/CplusplusProject/CplusplusProject.h File Reference</a>	33
6.5	<a href="#">CplusplusProject.h</a>	34
6.6	<a href="#">Source/CplusplusProject/Environment/AInteractableActor.cpp File Reference</a>	34
6.7	<a href="#">Source/CplusplusProject/Environment/AInteractableActor.h File Reference</a>	34
6.8	<a href="#">AInteractableActor.h</a>	34
6.9	<a href="#">Source/CplusplusProject/Environment/DoorOpenComponent.h File Reference</a>	35
6.10	<a href="#">DoorOpenComponent.h</a>	35
6.11	<a href="#">Source/CplusplusProject/Environment/FlickeringLight.cpp File Reference</a>	35
6.12	<a href="#">Source/CplusplusProject/Environment/FlickeringLight.h File Reference</a>	35
6.13	<a href="#">FlickeringLight.h</a>	36
6.14	<a href="#">Source/CplusplusProject/Environment/LockControllerComponent.h File Reference</a>	36
6.15	<a href="#">LockControllerComponent.h</a>	36
6.16	<a href="#">Source/CplusplusProject/Interfaces/InteractionInterface.h File Reference</a>	37
6.17	<a href="#">InteractionInterface.h</a>	37
6.18	<a href="#">Source/CplusplusProject/Player/MyPlayerController.cpp File Reference</a>	37
6.19	<a href="#">Source/CplusplusProject/Player/MyPlayerController.h File Reference</a>	38
6.20	<a href="#">MyPlayerController.h</a>	38
6.21	<a href="#">Source/CplusplusProject/Structs/ItemData.cpp File Reference</a>	39

6.22 Source/CplusplusProject/Structs/ItemData.h File Reference . . . . .	39
6.22.1 Enumeration Type Documentation . . . . .	39
6.22.1.1 ItemType . . . . .	39
6.23 ItemData.h . . . . .	40
6.24 Source/CplusplusProject/Structs/SimonData.cpp File Reference . . . . .	40
6.25 Source/CplusplusProject/Structs/SimonData.h File Reference . . . . .	40
6.25.1 Enumeration Type Documentation . . . . .	40
6.25.1.1 ButtonColors . . . . .	40
6.25.1.2 Notes . . . . .	41
6.26 SimonData.h . . . . .	41
6.27 Source/CplusplusProject/UI/Controllers/GameOverUIController.cpp File Reference . . . . .	42
6.28 Source/CplusplusProject/UI/Controllers/GameOverUIController.h File Reference . . . . .	42
6.29 GameOverUIController.h . . . . .	42
6.30 Source/CplusplusProject/UI/Controllers/GameUIController.cpp File Reference . . . . .	43
6.31 Source/CplusplusProject/UI/Controllers/GameUIController.h File Reference . . . . .	43
6.32 GameUIController.h . . . . .	43
6.33 Source/CplusplusProject/UI/Controllers/MainMenuUIController.cpp File Reference . . . . .	44
6.34 Source/CplusplusProject/UI/Controllers/MainMenuUIController.h File Reference . . . . .	44
6.35 MainMenuUIController.h . . . . .	45
6.36 Source/CplusplusProject/UI/Controllers/UIController.cpp File Reference . . . . .	45
6.37 Source/CplusplusProject/UI/Controllers/UIController.h File Reference . . . . .	45
6.38 UIController.h . . . . .	45
6.39 Source/CplusplusProject/UI/Widgets/GameOverUI.h File Reference . . . . .	46
6.40 GameOverUI.h . . . . .	46
6.41 Source/CplusplusProject/UI/Widgets/InteractionUI.h File Reference . . . . .	46
6.42 InteractionUI.h . . . . .	47
6.43 Source/CplusplusProject/UI/Widgets/LockUI.cpp File Reference . . . . .	47
6.44 Source/CplusplusProject/UI/Widgets/LockUI.h File Reference . . . . .	47
6.45 LockUI.h . . . . .	48
6.46 Source/CplusplusProject/UI/Widgets/MainMenuUI.cpp File Reference . . . . .	49
6.47 Source/CplusplusProject/UI/Widgets/MainMenuUI.h File Reference . . . . .	49
6.48 MainMenuUI.h . . . . .	49
<b>Index</b>	<b>51</b>





# Chapter 1

## CPlusProject

This project is a 2D RPG-style video game, developed using Unreal Engine 5.4.2 and C++.

The concept revolves around a music-based puzzle game in 2D, drawing inspiration from the childhood classic Simon.

A player awakens in an unfamiliar place, surrounded by thick walls and dim lighting. There is one door to exit the mystery room, but it is locked.

The player can make their way through several rooms, each room serving as a game level. To progress to the next one, the player must solve a puzzle placed on the door.

The lock plays a short melody. It contains 4 buttons (red, green, blue, yellow) that when pressed, emit a single note. To open the lock, the player must accurately replicate the short melody using these buttons.

For those struggling, a "hint" button is available which replays the melody.

The game's difficulty progressively increases, introducing variations such as changing each note button (so players will have to rely on their listening skills and not visual memory), altering the melody's scale, or adjusting the number of notes to be played.

To run the game open it in the Unreal Engine 5.4 editor, or email [reutgaming@gmail.com](mailto:reutgaming@gmail.com) for a platform-specific build. The scripts are located in the Source folder.



## Chapter 2

# Hierarchical Index

### 2.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

AActor	
AActorInteractableActor . . . . .	9
AFlickeringLight . . . . .	10
AHUD	
AUIController . . . . .	18
AGameOverUIController . . . . .	12
AGameUIController . . . . .	13
AMainMenuUIController . . . . .	15
APlayerController	
AMyPlayerController . . . . .	16
FItemData . . . . .	20
FSimonData . . . . .	21
IInteractionInterface . . . . .	22
AActorInteractableActor . . . . .	9
ModuleRules	
CplusplusProject . . . . .	19
UInterface	
UInteractionInterface . . . . .	24
UUserWidget	
UInteractionUI . . . . .	25
ULockUI . . . . .	25
UMainMenuUI . . . . .	30
UGameOverUI . . . . .	23



## Chapter 3

# Class Index

### 3.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

<a href="#">AAInteractableActor</a> . . . . .	9
<a href="#">AFlickeringLight</a> . . . . .	10
<a href="#">AGameOverUIController</a> . . . . .	12
<a href="#">AGameUIContoller</a> . . . . .	13
<a href="#">AMainMenuUIController</a> . . . . .	15
<a href="#">AMyPlayerController</a> . . . . .	16
<a href="#">AUIController</a> . . . . .	18
<a href="#">CplusProject</a> . . . . .	19
<a href="#">FItemData</a>	
Each intractable instance will have its own copy of an ItemData . . . . .	20
<a href="#">FSimonData</a> . . . . .	21
<a href="#">IInteractionInterface</a>	
Every object that the player can interact with must implement this interface . . . . .	22
<a href="#">UGameOverUI</a> . . . . .	23
<a href="#">UInteractionInterface</a>	
Unlike C++, in unreal you do NOT have to implement all functions in order to implement an interface . . . . .	24
<a href="#">UInteractionUI</a> . . . . .	25
<a href="#">ULockUI</a> . . . . .	25
<a href="#">UMainMenuUI</a> . . . . .	30



## Chapter 4

# File Index

### 4.1 File List

Here is a list of all files with brief descriptions:

Source/CplusplusProject/CplusplusProject.Build.cs	33
Source/CplusplusProject/CplusplusProject.cpp	33
Source/CplusplusProject/CplusplusProject.h	33
Source/CplusplusProject/Environment/AInteractableActor.cpp	34
Source/CplusplusProject/Environment/AInteractableActor.h	34
Source/CplusplusProject/Environment/DoorOpenComponent.h	35
Source/CplusplusProject/Environment/FlickeringLight.cpp	35
Source/CplusplusProject/Environment/FlickeringLight.h	35
Source/CplusplusProject/Environment/LockControllerComponent.h	36
Source/CplusplusProject/Interfaces/InteractionInterface.h	37
Source/CplusplusProject/Player/MyPlayerController.cpp	37
Source/CplusplusProject/Player/MyPlayerController.h	38
Source/CplusplusProject/Structs/ItemData.cpp	39
Source/CplusplusProject/Structs/ItemData.h	39
Source/CplusplusProject/Structs/SimonData.cpp	40
Source/CplusplusProject/Structs/SimonData.h	40
Source/CplusplusProject/UI/Controllers/GameOverUIController.cpp	42
Source/CplusplusProject/UI/Controllers/GameOverUIController.h	42
Source/CplusplusProject/UI/Controllers/GameUIController.cpp	43
Source/CplusplusProject/UI/Controllers/GameUIController.h	43
Source/CplusplusProject/UI/Controllers/MainMenuUIController.cpp	44
Source/CplusplusProject/UI/Controllers/MainMenuUIController.h	44
Source/CplusplusProject/UI/Controllers/UIController.cpp	45
Source/CplusplusProject/UI/Controllers/UIController.h	45
Source/CplusplusProject/UI/Widgets/GameOverUI.h	46
Source/CplusplusProject/UI/Widgets/InteractionUI.h	46
Source/CplusplusProject/UI/Widgets/LockUI.cpp	47
Source/CplusplusProject/UI/Widgets/LockUI.h	47
Source/CplusplusProject/UI/Widgets/MainMenuUI.cpp	49
Source/CplusplusProject/UI/Widgets/MainMenuUI.h	49





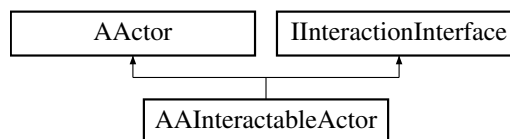
## Chapter 5

# Class Documentation

### 5.1 AAInteractableActor Class Reference

```
#include <AInteractableActor.h>
```

Inheritance diagram for AAInteractableActor:



#### Public Member Functions

- [AAInteractableActor](#) ()
- virtual void [BeginFocus2D](#) () override
- virtual void [EndFocus2D](#) () override
- virtual const [FItemData](#) & [GetItemData](#) () const override

#### Public Member Functions inherited from [IInteractionInterface](#)

#### Public Attributes

- [FItemData](#) [itemData](#)  
*public because I need to set the value in editor*

#### 5.1.1 Constructor & Destructor Documentation

##### 5.1.1.1 AAInteractableActor()

```
AAInteractableActor::AAInteractableActor ()
```

## 5.1.2 Member Function Documentation

### 5.1.2.1 BeginFocus2D()

```
void AAInteractableActor::BeginFocus2D () [override], [virtual]
```

Implements [InteractionInterface](#).

### 5.1.2.2 EndFocus2D()

```
void AAInteractableActor::EndFocus2D () [override], [virtual]
```

Implements [InteractionInterface](#).

### 5.1.2.3 GetItemData()

```
virtual const FItemData & AAInteractableActor::GetItemData () const [inline], [override],  
[virtual]
```

Implements [InteractionInterface](#).

## 5.1.3 Member Data Documentation

### 5.1.3.1 itemData

```
FItemData AAInteractableActor::itemData
```

public because I need to set the value in editor

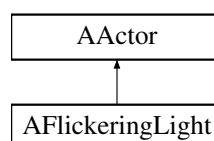
The documentation for this class was generated from the following files:

- Source/CplusplusProject/Environment/[AInteractableActor.h](#)
- Source/CplusplusProject/Environment/[AInteractableActor.cpp](#)

## 5.2 AFlickeringLight Class Reference

```
#include <FlickeringLight.h>
```

Inheritance diagram for AFlickeringLight:



### Public Member Functions

- [AFlickeringLight](#) ()
- virtual void [Tick](#) (float DeltaTime) override

### Public Attributes

- USpotLightComponent \* [spotlight](#)

### Protected Member Functions

- virtual void [BeginPlay](#) () override

## 5.2.1 Constructor & Destructor Documentation

### 5.2.1.1 AFlickeringLight()

```
AFlickeringLight::AFlickeringLight ()
```

## 5.2.2 Member Function Documentation

### 5.2.2.1 BeginPlay()

```
void AFlickeringLight::BeginPlay () [override], [protected], [virtual]
```

### 5.2.2.2 Tick()

```
void AFlickeringLight::Tick (  
    float DeltaTime) [override], [virtual]
```

## 5.2.3 Member Data Documentation

### 5.2.3.1 spotlight

```
USpotLightComponent* AFlickeringLight::spotlight
```

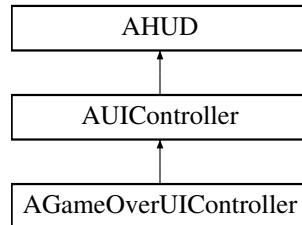
The documentation for this class was generated from the following files:

- Source/CplusplusProject/Environment/[FlickeringLight.h](#)
- Source/CplusplusProject/Environment/[FlickeringLight.cpp](#)

## 5.3 AGameOverUIController Class Reference

```
#include <GameOverUIController.h>
```

Inheritance diagram for AGameOverUIController:



### Public Attributes

- TSubclassOf< [UGameOverUI](#) > [gameOverUIClass](#)

### Protected Member Functions

- virtual void [BeginPlay](#) () override

### Protected Member Functions inherited from [AUIController](#)

- void [SwitchToUIControls](#) ()
- void [SwitchToPlayerControls](#) ()

### Additional Inherited Members

### Public Member Functions inherited from [AUIController](#)

- virtual void [CloseAll](#) ()

### Protected Attributes inherited from [AUIController](#)

- FInputModeUIOnly [inputModeUI](#)
- FInputModeGameOnly [gameInputMode](#)

## 5.3.1 Member Function Documentation

### 5.3.1.1 BeginPlay()

```
void AGameOverUIController::BeginPlay () [override], [protected], [virtual]
```

## 5.3.2 Member Data Documentation

### 5.3.2.1 gameOverUIClass

```
TSubclassOf<UGameOverUI> AGameOverUIContoller::gameOverUIClass
```

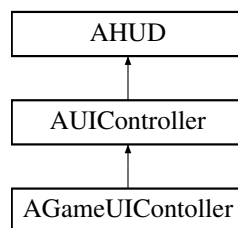
The documentation for this class was generated from the following files:

- Source/CplusplusProject/UI/Controllers/[GameOverUIContoller.h](#)
- Source/CplusplusProject/UI/Controllers/[GameOverUIContoller.cpp](#)

## 5.4 AGameUIContoller Class Reference

```
#include <GameUIContoller.h>
```

Inheritance diagram for AGameUIContoller:



### Public Member Functions

- void [UpdateInteractionUI](#) (const AActor \*interactedActor)
- virtual void [CloseAll](#) () override
- void [OpenEncourageInteractUI](#) ()
- void [CloseEncourageInteractUI](#) ()
- void [SetLockController](#) (ULockControllerComponent \*newController)
- void [CloseLockUI](#) ()

### Public Member Functions inherited from [AUIController](#)

#### Public Attributes

- TSubclassOf< [UInteractionUI](#) > [interactionUIClass](#)
- TSubclassOf< [UUserWidget](#) > [encourageInteractionUIClass](#)
- TSubclassOf< [ULockUI](#) > [lockUIClass](#)

### Protected Member Functions

- virtual void [BeginPlay](#) () override

## Protected Member Functions inherited from [AUIController](#)

- void [SwitchToUIControls](#) ()
- void [SwitchToPlayerControls](#) ()

## Additional Inherited Members

## Protected Attributes inherited from [AUIController](#)

- FInputModeUIOnly [inputModeUI](#)
- FInputModeGameOnly [gameInputMode](#)

## 5.4.1 Member Function Documentation

### 5.4.1.1 BeginPlay()

```
void AGameUIController::BeginPlay () [override], [protected], [virtual]
```

### 5.4.1.2 CloseAll()

```
void AGameUIController::CloseAll () [override], [virtual]
```

Reimplemented from [AUIController](#).

### 5.4.1.3 CloseEncourageInteractUI()

```
void AGameUIController::CloseEncourageInteractUI ()
```

### 5.4.1.4 CloseLockUI()

```
void AGameUIController::CloseLockUI ()
```

### 5.4.1.5 OpenEncourageInteractUI()

```
void AGameUIController::OpenEncourageInteractUI ()
```

### 5.4.1.6 SetLockController()

```
void AGameUIController::SetLockController (  
    ULockControllerComponent * newController)
```

#### 5.4.1.7 UpdateInteractionUI()

```
void AGameUIController::UpdateInteractionUI (
    const AActor * interactedActor)
```

### 5.4.2 Member Data Documentation

#### 5.4.2.1 encourageInteractionUIClass

```
TSubclassOf<UUserWidget> AGameUIController::encourageInteractionUIClass
```

#### 5.4.2.2 interactionUIClass

```
TSubclassOf<UInteractionUI> AGameUIController::interactionUIClass
```

#### 5.4.2.3 lockUIClass

```
TSubclassOf<ULockUI> AGameUIController::lockUIClass
```

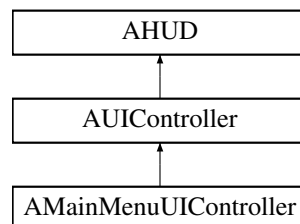
The documentation for this class was generated from the following files:

- Source/CplusplusProject/UI/Controllers/[GameUIController.h](#)
- Source/CplusplusProject/UI/Controllers/[GameUIController.cpp](#)

## 5.5 AMainMenuUIController Class Reference

```
#include <MainMenuUIController.h>
```

Inheritance diagram for AMainMenuUIController:



#### Public Attributes

- TSubclassOf< [UMainMenuUI](#) > [mainMenuClass](#)  
*needs to be chosen in the editor*

#### Protected Member Functions

- virtual void [BeginPlay](#) () override

## Protected Member Functions inherited from [AUIController](#)

- void [SwitchToUIControls](#) ()
- void [SwitchToPlayerControls](#) ()

## Additional Inherited Members

## Public Member Functions inherited from [AUIController](#)

- virtual void [CloseAll](#) ()

## Protected Attributes inherited from [AUIController](#)

- FInputModeUIOnly [inputModeUI](#)
- FInputModeGameOnly [gameInputMode](#)

## 5.5.1 Member Function Documentation

### 5.5.1.1 BeginPlay()

```
void AMainMenuUIController::BeginPlay () [override], [protected], [virtual]
```

## 5.5.2 Member Data Documentation

### 5.5.2.1 mainMenuClass

```
TSubclassOf<UMainMenuUI> AMainMenuUIController::mainMenuClass
```

needs to be chosen in the editor

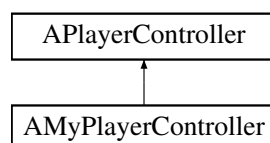
The documentation for this class was generated from the following files:

- Source/CplusplusProject/UI/Controllers/[MainMenuUIController.h](#)
- Source/CplusplusProject/UI/Controllers/[MainMenuUIController.cpp](#)

## 5.6 AMyPlayerController Class Reference

```
#include <MyPlayerController.h>
```

Inheritance diagram for AMyPlayerController:





## Public Member Functions

- [AMyPlayerController](#) ()
- void [SetLastInteractedActor](#) (AActor \*acotor)

## Public Attributes

- UInputAction \* [actionMove](#) = nullptr
- UInputAction \* [actionInteract](#) = nullptr
- UInputMappingContext \* [inputMappingContext](#) = nullptr
- FVector2D [direction](#)

## Protected Member Functions

- virtual void [OnPossess](#) (APawn \*pawn) override  
*on enable and on disable. pawn = player.*
- virtual void [OnUnPossess](#) () override

## 5.6.1 Constructor & Destructor Documentation

### 5.6.1.1 AMyPlayerController()

```
AMyPlayerController::AMyPlayerController ()
```

## 5.6.2 Member Function Documentation

### 5.6.2.1 OnPossess()

```
void AMyPlayerController::OnPossess (
    APawn * pawn) [override], [protected], [virtual]
```

on enable and on disable. pawn = player.

on enable and on disable. pawn = player. This gets called once the controller gets connected to the player that was spawned.

### 5.6.2.2 OnUnPossess()

```
void AMyPlayerController::OnUnPossess () [override], [protected], [virtual]
```

### 5.6.2.3 SetLastInteractedActor()

```
void AMyPlayerController::SetLastInteractedActor (
    AActor * acotor)
```

## 5.6.3 Member Data Documentation

### 5.6.3.1 actionInteract

```
UInputAction* AMyPlayerController::actionInteract = nullptr
```

### 5.6.3.2 actionMove

```
UInputAction* AMyPlayerController::actionMove = nullptr
```

### 5.6.3.3 direction

```
FVector2D AMyPlayerController::direction
```

### 5.6.3.4 inputMappingContext

```
UInputMappingContext* AMyPlayerController::inputMappingContext = nullptr
```

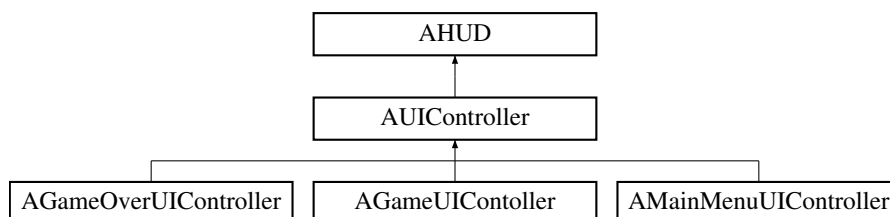
The documentation for this class was generated from the following files:

- Source/CplusplusProject/Player/[MyPlayerController.h](#)
- Source/CplusplusProject/Player/[MyPlayerController.cpp](#)

## 5.7 AUIController Class Reference

```
#include <UIController.h>
```

Inheritance diagram for AUIController:



### Public Member Functions

- virtual void [CloseAll](#) ()

### Protected Member Functions

- void [SwitchToUIControls](#) ()
- void [SwitchToPlayerControls](#) ()

### Protected Attributes

- FInputModeUIOnly [inputModeUI](#)
- FInputModeGameOnly [gameInputMode](#)

## 5.7.1 Member Function Documentation

### 5.7.1.1 CloseAll()

```
virtual void AUIController::CloseAll () [inline], [virtual]
```

Reimplemented in [AGameUIController](#).

### 5.7.1.2 SwitchToPlayerControls()

```
void AUIController::SwitchToPlayerControls () [protected]
```

### 5.7.1.3 SwitchToUIControls()

```
void AUIController::SwitchToUIControls () [protected]
```

## 5.7.2 Member Data Documentation

### 5.7.2.1 gameInputMode

```
FInputModeGameOnly AUIController::gameInputMode [protected]
```

### 5.7.2.2 inputModeUI

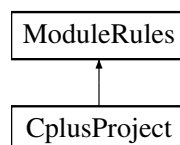
```
FInputModeUIOnly AUIController::inputModeUI [protected]
```

The documentation for this class was generated from the following files:

- Source/CplusplusProject/UI/Controllers/[UIController.h](#)
- Source/CplusplusProject/UI/Controllers/[UIController.cpp](#)

## 5.8 CplusplusProject Class Reference

Inheritance diagram for CplusplusProject:



## Public Member Functions

- [CplusplusProject](#) (ReadOnlyTargetRules Target)

## 5.8.1 Constructor & Destructor Documentation

### 5.8.1.1 CplusplusProject()

```
CplusplusProject.CplusplusProject (  
    ReadOnlyTargetRules Target) [inline]
```

The documentation for this class was generated from the following file:

- Source/CplusplusProject/[CplusplusProject.Build.cs](#)

## 5.9 FItemData Struct Reference

each intractable instance will have its own copy of an ItemData

```
#include <ItemData.h>
```

## Public Member Functions

- [GENERATED\\_BODY](#) ()
- [FItemData](#) ()

## Public Attributes

- [ItemType](#) itemType
- [FText](#) description
- [bool](#) interactable

## 5.9.1 Detailed Description

each intractable instance will have its own copy of an ItemData

## 5.9.2 Constructor & Destructor Documentation

### 5.9.2.1 FItemData()

```
FItemData::FItemData ()
```

## 5.9.3 Member Function Documentation

### 5.9.3.1 GENERATED\_BODY()

```
FItemData::GENERATED_BODY ()
```

## 5.9.4 Member Data Documentation

### 5.9.4.1 description

```
FText FItemData::description
```

### 5.9.4.2 interactable

```
bool FItemData::interactable
```

### 5.9.4.3 itemType

```
ItemType FItemData::itemType
```

The documentation for this struct was generated from the following files:

- Source/CplusplusProject/Structs/[ItemData.h](#)
- Source/CplusplusProject/Structs/[ItemData.cpp](#)

## 5.10 FSimonData Struct Reference

```
#include <SimonData.h>
```

### Public Member Functions

- [GENERATED\\_BODY](#) ()
- [FSimonData](#) ()

### Public Attributes

- TMap< [ButtonColors](#), [Notes](#) > [mapColorToNote](#)  
*indicates which node will sound when a button is pressed audios that should be heard when each button is pressed*
- TArray< [Notes](#) > [sequence](#)  
*the unique sequence of notes the device shall play*

## 5.10.1 Constructor & Destructor Documentation

### 5.10.1.1 FSimonData()

```
FSimonData::FSimonData ()
```

## 5.10.2 Member Function Documentation

### 5.10.2.1 GENERATED\_BODY()

```
FSimonData::GENERATED_BODY ()
```

## 5.10.3 Member Data Documentation

### 5.10.3.1 mapColorToNote

```
TMap<ButtonColors, Notes> FSimonData::mapColorToNote
```

indicates which node will sound when a button is pressed audios that should be heard when each button is pressed

### 5.10.3.2 sequence

```
TArray<Notes> FSimonData::sequence
```

the unique sequence of notes the device shall play

The documentation for this struct was generated from the following files:

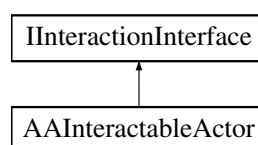
- Source/CplusplusProject/Structs/[SimonData.h](#)
- Source/CplusplusProject/Structs/[SimonData.cpp](#)

## 5.11 IInteractionInterface Class Reference

Every object that the player can interact with must implement this interface.

```
#include <InteractionInterface.h>
```

Inheritance diagram for IInteractionInterface:



**Public Member Functions**

- virtual void [BeginFocus2D](#) ()=0
- virtual void [EndFocus2D](#) ()=0
- virtual const [FItemData](#) & [GetItemData](#) () const =0

**5.11.1 Detailed Description**

Every object that the player can interact with must implement this interface.

**5.11.2 Member Function Documentation****5.11.2.1 BeginFocus2D()**

```
virtual void IInteractionInterface::BeginFocus2D () [pure virtual]
```

Implemented in [AAInteractableActor](#).

**5.11.2.2 EndFocus2D()**

```
virtual void IInteractionInterface::EndFocus2D () [pure virtual]
```

Implemented in [AAInteractableActor](#).

**5.11.2.3 GetItemData()**

```
virtual const FItemData & IInteractionInterface::GetItemData () const [pure virtual]
```

Implemented in [AAInteractableActor](#).

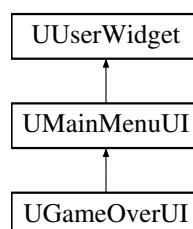
The documentation for this class was generated from the following file:

- Source/CplusplusProject/Interfaces/[InteractionInterface.h](#)

**5.12 UGameOverUI Class Reference**

```
#include <GameOverUI.h>
```

Inheritance diagram for UGameOverUI:



### Additional Inherited Members

### Public Member Functions inherited from [UMainMenuUI](#)

- void [StartGame](#) ()

### Public Attributes inherited from [UMainMenuUI](#)

- UButton \* [startGameButton](#)

### Protected Member Functions inherited from [UMainMenuUI](#)

The documentation for this class was generated from the following file:

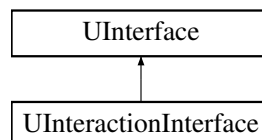
- Source/CplusplusProject/UI/Widgets/[GameOverUI.h](#)

## 5.13 UInteractionInterface Class Reference

Unlike C++, in unreal you do NOT have to implement all functions in order to implement an interface.

```
#include <InteractionInterface.h>
```

Inheritance diagram for UInteractionInterface:



### 5.13.1 Detailed Description

Unlike C++, in unreal you do NOT have to implement all functions in order to implement an interface.

This class does not need to be modified.

The documentation for this class was generated from the following file:

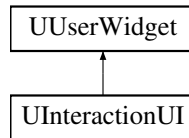
- Source/CplusplusProject/Interfaces/[InteractionInterface.h](#)



## 5.14 UInteractionUI Class Reference

```
#include <InteractionUI.h>
```

Inheritance diagram for UInteractionUI:



### Public Member Functions

- void [UpdateWidget](#) (const FText &data) const

### 5.14.1 Member Function Documentation

#### 5.14.1.1 UpdateWidget()

```
void UInteractionUI::UpdateWidget (  
    const FText & data) const [inline]
```

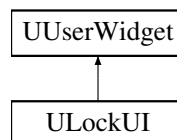
The documentation for this class was generated from the following file:

- Source/CplusplusProject/UI/Widgets/[InteractionUI.h](#)

## 5.15 ULockUI Class Reference

```
#include <LockUI.h>
```

Inheritance diagram for ULockUI:



## Public Member Functions

- void [BindButtons](#) ()
- void [RedButton](#) ()
- void [BlueButton](#) ()
- void [GreenButton](#) ()
- void [YellowButton](#) ()
- void [GiveHint](#) ()
- void [OnOpen](#) ()
  - when the UI panel gets opened*
- void [OnUnlocked](#) ()
  - when the player successfully repeated the sequence*
- void [OnClose](#) ()
  - when the UI panel gets closed*
- void [PlaySequence](#) ()
- bool [CheckSequence](#) ()
  - returns true if the player was able to play the sequence that is set in the lock data*
- void [SetLockController](#) (ULockControllerComponent \*newController)

## Public Attributes

- UIButton \* [redButton](#)
- UIButton \* [greenButton](#)
- UIButton \* [blueButton](#)
- UIButton \* [yellowButton](#)
- UIButton \* [hintButton](#)
- UIButton \* [closeButton](#)
- UIImage \* [errorImage](#)
- UIImage \* [greenHighlightImage](#)
- UIImage \* [blueHighlightImage](#)
- UIImage \* [yellowHighlightImage](#)
- UIImage \* [redHighlightImage](#)
- TArray< USoundBase \* > [notesAudio](#)
- USoundBase \* [errorSound](#)
- float [timeBetweenNotes](#) = 0.7f
- float [showButtonHightlightTime](#) = 0.7f
- float [showErrorTime](#) = 1.3f
- float [volumeMultiplier](#) = 1.5
- TSubclassOf< [AAInteractableActor](#) > [openDoorActorClass](#)

## Protected Member Functions

- virtual void [NativeConstruct](#) () override

## 5.15.1 Member Function Documentation

### 5.15.1.1 BindButtons()

```
void ULockUI::BindButtons ()
```

### 5.15.1.2 BlueButton()

```
void ULockUI::BlueButton ()
```

### 5.15.1.3 CheckSequence()

```
bool ULockUI::CheckSequence ()
```

returns true if the player was able to play the sequence that is set in the lock data  
array is bigger, or equal but not the correct sequence

### 5.15.1.4 GiveHint()

```
void ULockUI::GiveHint ()
```

### 5.15.1.5 GreenButton()

```
void ULockUI::GreenButton ()
```

### 5.15.1.6 NativeConstruct()

```
void ULockUI::NativeConstruct () [override], [protected], [virtual]
```

### 5.15.1.7 OnClose()

```
void ULockUI::OnClose ()
```

when the UI panel gets closed

### 5.15.1.8 OnOpen()

```
void ULockUI::OnOpen ()
```

when the UI panel gets opened

### 5.15.1.9 OnUnlocked()

```
void ULockUI::OnUnlocked ()
```

when the player successfully repeated the sequence

generate an open door class instance in the exact same the locked door was before the open door class inherits from intractable actor, so player can leave the room destroy it since we left the lock and it is no longer relevant

get the scene name that was set on the lock controller, which says which scene should be loaded once the lock is open

tell the UI panel this data is what it needs to interact with now

#### 5.15.1.10 PlaySequence()

```
void ULockUI::PlaySequence ()
```

#### 5.15.1.11 RedButton()

```
void ULockUI::RedButton ()
```

#### 5.15.1.12 SetLockController()

```
void ULockUI::SetLockController (
    ULockControllerComponent * newController)
```

#### 5.15.1.13 YellowButton()

```
void ULockUI::YellowButton ()
```

### 5.15.2 Member Data Documentation

#### 5.15.2.1 blueButton

```
UIButton* ULockUI::blueButton
```

#### 5.15.2.2 blueHighlightImage

```
UIImage* ULockUI::blueHighlightImage
```

#### 5.15.2.3 closeButton

```
UIButton* ULockUI::closeButton
```

#### 5.15.2.4 errorImage

```
UIImage* ULockUI::errorImage
```

#### 5.15.2.5 errorSound

```
USoundBase* ULockUI::errorSound
```

#### 5.15.2.6 greenButton

```
UIButton* ULockUI::greenButton
```

### 5.15.2.7 greenHighlightImage

```
UIImage* ULockUI::greenHighlightImage
```

### 5.15.2.8 hintButton

```
UIButton* ULockUI::hintButton
```

### 5.15.2.9 notesAudio

```
TArray<USoundBase*> ULockUI::notesAudio
```

### 5.15.2.10 openDoorActorClass

```
TSubclassOf<AAInteractableActor> ULockUI::openDoorActorClass
```

### 5.15.2.11 redButton

```
UIButton* ULockUI::redButton
```

### 5.15.2.12 redHighlightImage

```
UIImage* ULockUI::redHighlightImage
```

### 5.15.2.13 showButtonHighlightTime

```
float ULockUI::showButtonHighlightTime = 0.7f
```

### 5.15.2.14 showErrorTime

```
float ULockUI::showErrorTime = 1.3f
```

### 5.15.2.15 timeBetweenNotes

```
float ULockUI::timeBetweenNotes = 0.7f
```

### 5.15.2.16 volumeMultiplier

```
float ULockUI::volumeMultiplier = 1.5
```

### 5.15.2.17 yellowButton

```
UIButton* ULockUI::yellowButton
```

### 5.15.2.18 yellowHighlightImage

```
UIImage* ULockUI::yellowHighlightImage
```

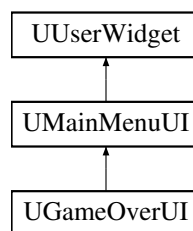
The documentation for this class was generated from the following files:

- Source/CplusplusProject/UI/Widgets/[LockUI.h](#)
- Source/CplusplusProject/UI/Widgets/[LockUI.cpp](#)

## 5.16 UMainMenuUI Class Reference

```
#include <MainMenuUI.h>
```

Inheritance diagram for UMainMenuUI:



### Public Member Functions

- void [StartGame](#) ()

### Public Attributes

- UIButton \* [startGameButton](#)

### Protected Member Functions

- virtual void [NativeConstruct](#) () override

## 5.16.1 Member Function Documentation

### 5.16.1.1 NativeConstruct()

```
void UMainMenuUI::NativeConstruct () [override], [protected], [virtual]
```

### 5.16.1.2 StartGame()

```
void UMainMenuUI::StartGame ()
```

## 5.16.2 Member Data Documentation

### 5.16.2.1 startGameButton

```
UButton* UMainMenuUI::startGameButton
```

The documentation for this class was generated from the following files:

- Source/CplusplusProject/UI/Widgets/[MainMenuUI.h](#)
- Source/CplusplusProject/UI/Widgets/[MainMenuUI.cpp](#)





## Chapter 6

# File Documentation

### 6.1 README.txt File Reference

### 6.2 Source/CplusplusProject/CplusplusProject.Build.cs File Reference

#### Classes

- class [CplusplusProject](#)

### 6.3 Source/CplusplusProject/CplusplusProject.cpp File Reference

```
#include "CplusplusProject.h"  
#include "Modules/ModuleManager.h"
```

#### Functions

- [IMPLEMENT\\_PRIMARY\\_GAME\\_MODULE](#) (FDefaultGameModuleImpl, [CplusplusProject](#), "CplusplusProject")

#### 6.3.1 Function Documentation

##### 6.3.1.1 IMPLEMENT\_PRIMARY\_GAME\_MODULE()

```
IMPLEMENT_PRIMARY_GAME_MODULE (  
    FDefaultGameModuleImpl ,  
    CplusplusProject ,  
    "CplusplusProject" )
```

### 6.4 Source/CplusplusProject/CplusplusProject.h File Reference

```
#include "CoreMinimal.h"
```

## 6.5 CplusplusProject.h

[Go to the documentation of this file.](#)

```
00001 // Fill out your copyright notice in the Description page of Project Settings.
00002
00003 #pragma once
00004
00005 #include "CoreMinimal.h"
00006
```

## 6.6 Source/CplusplusProject/Environment/AInteractableActor.cpp File Reference

```
#include "AInteractableActor.h"
```

## 6.7 Source/CplusplusProject/Environment/AInteractableActor.h File Reference

```
#include "CoreMinimal.h"
#include "GameFramework/Actor.h"
#include "../Interfaces/InteractionInterface.h"
#include "AInteractableActor.generated.h"
```

### Classes

- class [AAInteractableActor](#)

## 6.8 AInteractableActor.h

[Go to the documentation of this file.](#)

```
00001 #pragma once
00002
00003 #include "CoreMinimal.h"
00004 #include "GameFramework/Actor.h"
00005 #include "../Interfaces/InteractionInterface.h"
00006 #include "AInteractableActor.generated.h"
00007
00008 UCLASS()
00009 class CPLUSPROJECT_API AAInteractableActor : public AActor, public IInteractionInterface
00010 {
00011     GENERATED_BODY()
00012
00013 public:
00014     // ===== FUNCTIONS =====
00015     AAInteractableActor();
00016     virtual void BeginFocus2D() override;
00017     virtual void EndFocus2D() override;
00018     virtual const FItemData& GetItemData() const override{ return itemData; };
00019     // ===== PROPERTIES =====
00020     UPROPERTY(editanywhere)
00021     FItemData itemData;
00022 };
00023
```

## 6.9 Source/CplusplusProject/Environment/DoorOpenComponent.h File Reference

```
#include "CoreMinimal.h"
#include "Components/ActorComponent.h"
#include "Kismet/GameplayStatics.h"
#include "DoorOpenComponent.generated.h"
```

## 6.10 DoorOpenComponent.h

[Go to the documentation of this file.](#)

```
00001 #pragma once
00002
00003 #include "CoreMinimal.h"
00004 #include "Components/ActorComponent.h"
00005 #include "Kismet/GameplayStatics.h"
00006 #include "DoorOpenComponent.generated.h"
00007
00008 UCLASS( ClassGroup=(Custom), meta=(BlueprintSpawnableComponent) )
00009 class CPLUSPROJECT_API UDoorOpenComponent : public UActorComponent
00010 {
00011     GENERATED_BODY()
00012
00013 private:
00014     UPROPERTY()
00016     FString nextScene;
00017
00018 public:
00019     inline void OpenDoor()
00020     {
00021         if (nextScene.IsEmpty())
00022         {
00023             UE_LOG(LogTemp, Error, TEXT("Trying to open door but next scene name is empty!
00024             DOrOpenComponent"));
00025             return;
00026         }
00027         UGameplayStatics::OpenLevel(GetWorld(), FName(nextScene));
00028     }
00029     inline void SetSceneName(const FString& sceneName) { nextScene = sceneName; }
00030 };
```

## 6.11 Source/CplusplusProject/Environment/FlickeringLight.cpp File Reference

```
#include "FlickeringLight.h"
```

## 6.12 Source/CplusplusProject/Environment/FlickeringLight.h File Reference

```
#include "CoreMinimal.h"
#include "GameFramework/Actor.h"
#include "Components/SpotLightComponent.h"
#include "FlickeringLight.generated.h"
```

## Classes

- class [AFlickeringLight](#)

## 6.13 FlickeringLight.h

[Go to the documentation of this file.](#)

```
00001 #pragma once
00002
00003 #include "CoreMinimal.h"
00004 #include "GameFramework/Actor.h"
00005 #include "Components/SpotLightComponent.h"
00006 #include "FlickeringLight.generated.h"
00007
00008 UCLASS()
00009 class CPLUSPROJECT_API AFlickeringLight : public AActor
00010 {
00011     GENERATED_BODY()
00012
00013 public:
00014     // ===== FUNCTIONS =====
00015     AFlickeringLight();
00016     virtual void Tick(float DeltaTime) override;
00017     // ===== PROPERTIES =====
00018     UPROPERTY(EditAnywhere, Category = "Light Control")
00019     USpotLightComponent* spotlight;
00020 protected:
00021     virtual void BeginPlay() override;
00022 private:
00023     // ===== PROPERTIES =====
00024     UPROPERTY(EditAnywhere, Category = "Flicker Settings")
00025     float minIntensity;
00026
00027     UPROPERTY(EditAnywhere, Category = "Flicker Settings")
00028     float maxIntensity;
00029
00030     UPROPERTY(EditAnywhere, Category = "Flicker Settings")
00031     float flickerBreak;
00033     float timePassed;
00034     // ===== FUNCTIONS =====
00035     void FlickerLight();
00036
00037 };
```

## 6.14 Source/CplusplusProject/Environment/LockControllerComponent.h File Reference

```
#include "CoreMinimal.h"
#include "Components/ActorComponent.h"
#include "../Structs/SimonData.h"
#include "LockControllerComponent.generated.h"
```

## 6.15 LockControllerComponent.h

[Go to the documentation of this file.](#)

```
00001 #pragma once
00002
00003 #include "CoreMinimal.h"
00004 #include "Components/ActorComponent.h"
00005 #include "../Structs/SimonData.h"
00006 #include "LockControllerComponent.generated.h"
00007
00008
00009 UCLASS(ClassGroup = (Custom), meta = (BlueprintSpawnableComponent))
```

```

00010 class CPLUSPROJECT_API ULockControllerComponent : public UActorComponent
00011 {
00012     GENERATED_BODY()
00013
00014 public:
00015     UPROPERTY(EditAnywhere, Category = "Lock Data")
00016     FSimonData simonData;
00017     UPROPERTY(EditAnywhere, Category = "Lock Data")
00018     FString nextLevelName;
00019
00020
00021 };

```

## 6.16 Source/CplusplusProject/Interfaces/InteractionInterface.h File Reference

```

#include "CoreMinimal.h"
#include "UObject/Interface.h"
#include "../Structs/ItemData.h"
#include "InteractionInterface.generated.h"

```

### Classes

- class [UInteractionInterface](#)  
*Unlike C++, in unreal you do NOT have to implement all functions in order to implement an interface.*
- class [IInteractionInterface](#)  
*Every object that the player can interact with must implement this interface.*

## 6.17 InteractionInterface.h

[Go to the documentation of this file.](#)

```

00001
00002
00003 #pragma once
00004
00005 #include "CoreMinimal.h"
00006 #include "UObject/Interface.h"
00007 #include "../Structs/ItemData.h"
00008 #include "InteractionInterface.generated.h"
00009
00010 UINTERFACE(MinimalAPI)
00011 class UInteractionInterface : public UInterface
00012 {
00013     GENERATED_BODY()
00014 };
00015
00016 class CPLUSPROJECT_API IInteractionInterface
00017 {
00018     GENERATED_BODY()
00019 public:
00020     // ===== FUNCTIONS =====
00021     virtual void BeginFocus2D()=0;
00022     virtual void EndFocus2D() = 0;
00023     virtual const FItemData& GetItemData() const = 0;
00024 };

```

## 6.18 Source/CplusplusProject/Player/MyPlayerController.cpp File Reference

```

#include "MyPlayerController.h"
#include "EnhancedInputSubsystems.h"

```

## 6.19 Source/CplusplusProject/Player/MyPlayerController.h File Reference

```
#include "CoreMinimal.h"
#include "Components/CapsuleComponent.h"
#include "GameFramework/PlayerController.h"
#include "EnhancedInputComponent.h"
#include "InputMappingContext.h"
#include "GameFramework/Character.h"
#include "Delegates/Delegate.h"
#include "../UI/Controllers/GameUIController.h"
#include "../Interfaces/InteractionInterface.h"
#include "MyPlayerController.generated.h"
```

### Classes

- class [AMyPlayerController](#)

## 6.20 MyPlayerController.h

[Go to the documentation of this file.](#)

```
00001 #pragma once
00002
00003 #include "CoreMinimal.h"
00004 #include "Components/CapsuleComponent.h"
00005 #include "GameFramework/PlayerController.h"
00006 #include "EnhancedInputComponent.h"
00007 #include "InputMappingContext.h"
00008 #include "GameFramework/Character.h"
00009 #include "Delegates/Delegate.h"
00010 #include "../UI/Controllers/GameUIController.h"
00011 #include "../Interfaces/InteractionInterface.h"
00012 #include "MyPlayerController.generated.h"
00013
00014 UCLASS(Abstract)
00015 class CPLUSPROJECT_API AMyPlayerController : public APlayerController
00016 {
00017     GENERATED_BODY()
00018
00019 public:
00020
00021     // ===== PROPERTIES =====
00022     UPROPERTY(EditDefaultsOnly, BlueprintReadOnly, Category = "Player Input")
00023     UInputAction* actionMove= nullptr;
00024     UPROPERTY(EditDefaultsOnly, BlueprintReadOnly, Category = "Player Input")
00025     UInputAction* actionInteract= nullptr;
00026     UPROPERTY(EditDefaultsOnly, BlueprintReadOnly, Category = "Player Input")
00027     UInputMappingContext* inputMappingContext = nullptr;
00028     UPROPERTY(BlueprintReadWrite, VisibleAnywhere)
00029     FVector2D direction;
00030     // ===== FUNCTIONS =====
00031     AMyPlayerController();
00032     UFUNCTION()
00033     void SetLastInteractedActor(AActor* acotor);
00034 protected:
00035     // ===== FUNCTIONS =====
00037     virtual void OnPossess(APawn* pawn) override;
00038     virtual void OnUnPossess() override;
00039 private:
00040     // ===== PROPERTIES =====
00041
00042     UPROPERTY()
00043     UEnhancedInputComponent* inputComponentPtr = nullptr;
00044     UPROPERTY()
00045     ACharacter* playerPtr = nullptr;
00046     UPROPERTY()
00047     UCapsuleComponent* playerCollisionComponent = nullptr;
00048     UPROPERTY()
00049     AGameUIController* uiController= nullptr;
```

```

00051     UPROPERTY()
00052     AActor* lastInteractedActor = nullptr;
00053     float movementSpeed = 100.0;
00055     bool canInteract;
00057     bool isInteracting;
00058 // ===== FUNCTIONS =====
00059 //will be called on collision enter. here I will check if I hit anything that inherits from
    InteractionInterface
00060     UFUNCTION()
00061     void OverlapBegin(UPrimitiveComponent* OverlappedComponent, AActor* OtherActor,
    UPrimitiveComponent* OtherComp, int32 OtherBodyIndex, bool bFromSweep, const FHitResult& SweepResult);
00062     UFUNCTION()
00063     void OverlapEnd(UPrimitiveComponent* OverlappedComponent, AActor* OtherActor, UPrimitiveComponent*
    OtherComp, int32 OtherBodyIndex);
00065     void HandleInteractInput();
00066     void HandleMoveInput(const FInputActionValue& inputActionValue);
00067 };

```

## 6.21 Source/CplusplusProject/Structs/ItemData.cpp File Reference

```
#include "ItemData.h"
```

## 6.22 Source/CplusplusProject/Structs/ItemData.h File Reference

```

#include "CoreMinimal.h"
#include "ItemData.generated.h"

```

### Classes

- struct [FItemData](#)  
each intractable instance will have its own copy of an ItemData

### Enumerations

- enum class [ItemType](#) : uint8 {  
    [Interactable](#) , [Static](#) , [Lock](#) , [OpenDoor](#) ,  
    [NPC](#) , [Pickup](#) }

### 6.22.1 Enumeration Type Documentation

#### 6.22.1.1 ItemType

```
enum class ItemType : uint8 [strong]
```

#### Enumerator

Interactable	
Static	
Lock	
OpenDoor	
NPC	
Pickup	

## 6.23 ItemData.h

[Go to the documentation of this file.](#)

```
00001 #pragma once
00002
00003 #include "CoreMinimal.h"
00004 #include "ItemData.generated.h"
00005 UENUM()
00006 enum class ItemType : uint8
00007 {
00008     Interactable ,
00009     Static,
00010     Lock,
00011     OpenDoor,
00012     // for further development use
00013     NPC,
00014     Pickup,
00015 };
00016
00017 USTRUCT()
00019 struct CPLUSPROJECT_API FItemData
00020 {
00021     GENERATED_BODY();
00022 public:
00023     FItemData();
00024     UPROPERTY(editanywhere)
00025     ItemType itemType;
00026     UPROPERTY(editanywhere)
00027     FText description;
00028     UPROPERTY(editanywhere)
00029     bool interactable;
00030 };
```

## 6.24 Source/CplusplusProject/Structs/SimonData.cpp File Reference

```
#include "SimonData.h"
```

## 6.25 Source/CplusplusProject/Structs/SimonData.h File Reference

```
#include "CoreMinimal.h"
#include "SimonData.generated.h"
```

### Classes

- struct [FSimonData](#)

### Enumerations

- enum class [ButtonColors](#) : uint8 {  
    [Red](#) , [Green](#) , [Blue](#) , [Yellow](#) ,  
    [None](#) }
- enum class [Notes](#) : uint8 {  
    [A](#) , [Ab](#) , [B](#) , [Bb](#) ,  
    [C](#) , [D](#) , [Db](#) , [E](#) ,  
    [Eb](#) , [F](#) , [G](#) , [Gb](#) }

### 6.25.1 Enumeration Type Documentation

#### 6.25.1.1 ButtonColors

```
enum class ButtonColors : uint8 [strong]
```



## Enumerator

Red	
Green	
Blue	
Yellow	
None	

## 6.25.1.2 Notes

```
enum class Notes : uint8 [strong]
```

## Enumerator

A	
Ab	
B	
Bb	
C	
D	
Db	
E	
Eb	
F	
G	
Gb	

## 6.26 SimonData.h

[Go to the documentation of this file.](#)

```
00001 #pragma once
00002 #include "CoreMinimal.h"
00003 #include "SimonData.generated.h"
00004
00005 UENUM()
00006 enum class ButtonColors : uint8
00007 {
00008     Red,
00009     Green,
00010     Blue,
00011     Yellow,
00012     None
00013 };
00014 UENUM()
00015 enum class Notes : uint8
00016 {
00017     A,
00018     Ab,
00019     B,
00020     Bb,
00021     C,
00022     D,
00023     Db,
00024     E,
00025     Eb,
00026     F,
00027     G,
00028     Gb
00029 };
00030
```

```

00031 USTRUCT()
00032 struct CPLUSPROJECT_API FSimonData
00033 {
00034     GENERATED_BODY();
00035 public:
00036     FSimonData();
00039     UPROPERTY(EditAnywhere, Category = "Notes")
00040     TMap<ButtonColors, Notes> mapColorToNote;
00042     UPROPERTY(EditAnywhere, Category = "Sequence")
00043     TArray<Notes> sequence;
00044 };
00045
00046 // already exists
00047 /*FORCEINLINE bool operator==(const TArray<Notes>& Array1, const TArray<Notes>& Array2)
00048 {
00049     if (Array1.Num() != Array2.Num())
00050     {
00051         return false;
00052     }
00053
00054     for (int32 i = 0; i < Array1.Num(); i++)
00055     {
00056         if (Array1[i] != Array2[i])
00057         {
00058             return false;
00059         }
00060     }
00061
00062     return true;
00063 }*/

```

## 6.27 Source/CplusplusProject/UI/Controllers/GameOverUIController.cpp File Reference

```
#include "GameOverUIController.h"
```

## 6.28 Source/CplusplusProject/UI/Controllers/GameOverUIController.h File Reference

```

#include "CoreMinimal.h"
#include "GameFramework/HUD.h"
#include "CplusplusProject/UI/Widgets/GameOverUI.h"
#include "UIController.h"
#include "GameOverUIController.generated.h"

```

### Classes

- class [AGameOverUIController](#)

## 6.29 GameOverUIController.h

[Go to the documentation of this file.](#)

```

00001 #pragma once
00002
00003 #include "CoreMinimal.h"
00004 #include "GameFramework/HUD.h"
00005 #include "CplusplusProject/UI/Widgets/GameOverUI.h"
00006 #include "UIController.h"

```

```

00007 #include "GameOverUIController.generated.h"
00008
00009 UCLASS()
00010 class CPLUSPROJECT_API AGameOverUIController : public AUIController
00011 {
00012     GENERATED_BODY()
00013 public:
00014     // ===== PROPERTIES =====
00015     UPROPERTY(EditDefaultsOnly, Category = "Panels")
00016     TSubclassOf<UGameOverUI> gameOverUIClass;
00017     // ===== FUNCTIONS =====
00018 protected:
00019     virtual void BeginPlay() override;
00020
00021 };

```

## 6.30 Source/CplusplusProject/UI/Controllers/GameUIContoller.cpp File Reference

```

#include "GameUIContoller.h"
#include "CplusplusProject/Interfaces/InteractionInterface.h"
#include "CplusplusProject/Environment/DoorOpenComponent.h"

```

## 6.31 Source/CplusplusProject/UI/Controllers/GameUIContoller.h File Reference

```

#include "CoreMinimal.h"
#include "GameFramework/HUD.h"
#include "CplusplusProject/UI/Widgets/LockUI.h"
#include "CplusplusProject/Structs/ItemData.h"
#include "CplusplusProject/UI/Widgets/InteractionUI.h"
#include "UIController.h"
#include "GameUIContoller.generated.h"

```

### Classes

- class [AGameUIContoller](#)

## 6.32 GameUIContoller.h

[Go to the documentation of this file.](#)

```

00001 #pragma once
00002
00003 #include "CoreMinimal.h"
00004 #include "GameFramework/HUD.h"
00005 #include "CplusplusProject/UI/Widgets/LockUI.h"
00006 #include "CplusplusProject/Structs/ItemData.h"
00007 #include "CplusplusProject/UI/Widgets/InteractionUI.h"
00008 #include "UIController.h"
00009 #include "GameUIContoller.generated.h"
00010
00011 UCLASS()
00012 class CPLUSPROJECT_API AGameUIContoller : public AUIController
00013 {
00014     GENERATED_BODY()
00015 public:

```

```

00016 // ===== PROPERTIES =====
00017 // using TSubclassOf and not pointers because I want to refer to an instance of a blueprint class
// of a type (these are considered Blueprint classes, not instances of a class for some reason), and not
// to a class object that was created at runtime
00018 // I will be instantiating it in begin play
00019 UPROPERTY(EditDefaultsOnly, Category = "Panels")
00020 TSubclassOf<UInteractionUI> interactionUIClass;
00021 UPROPERTY(EditDefaultsOnly, Category = "Panels")
00022 TSubclassOf<UUserWidget> encourageInteractionUIClass;
00023 UPROPERTY(EditDefaultsOnly, Category = "Panels")
00024 TSubclassOf<ULockUI> lockUIClass;
00025
00026 // ===== FUNCTIONS =====
00027 void UpdateInteractionUI(const AActor* interactedActor);
00028 virtual void CloseAll() override;
00029 // when the player gets close enough to an intractable object, i.e. this object is in focus, then
// a popup will appear to let the player know it can interact with it.
00030 void OpenEncourageInteractUI();
00031 void CloseEncourageInteractUI();
00032 void SetLockController(ULockControllerComponent* newController);
00033 UFUNCTION()
00034 void CloseLockUI();
00035 protected:
00036 virtual void BeginPlay() override;
00037 private:
00038
00039 // ===== PROPERTIES =====
00040 UPROPERTY()
00041 UInteractionUI* interactionPanel;
00042 UPROPERTY()
00043 ULockUI* lockPanel;
00044 UPROPERTY()
00045 UUserWidget* encourageInteractionPanel;
00046 // ===== FUNCTIONS =====
00047 void CloseInteractionUI();
00048 void OpenInteractionUI();
00049 void OpenLockUI();
00050 };

```

### 6.33 Source/CplusplusProject/UI/Controllers/MainMenuUIController.cpp File Reference

```
#include "MainMenuUIController.h"
```

### 6.34 Source/CplusplusProject/UI/Controllers/MainMenuUIController.h File Reference

```

#include "CoreMinimal.h"
#include "GameFramework/HUD.h"
#include "UIController.h"
#include "CplusplusProject/UI/Widgets/GameOverUI.h"
#include "MainMenuUIController.generated.h"

```

#### Classes

- class [AMainMenuUIController](#)

## 6.35 MainMenuUIController.h

[Go to the documentation of this file.](#)

```
00001 #pragma once
00002
00003 #include "CoreMinimal.h"
00004 #include "GameFramework/HUD.h"
00005 #include "UIController.h"
00006 #include "CplusplusProject/UI/Widgets/GameOverUI.h"
00007 #include "MainMenuUIController.generated.h"
00008
00009 UCLASS()
00010 class CPLUSPROJECT_API AMainMenuUIController : public AUIController
00011 {
00012     GENERATED_BODY()
00013 public:
00014     // ===== PROPERTIES =====
00015     UPROPERTY(EditDefaultsOnly, Category = "Panels")
00016     TSubclassOf<UMainMenuUI> MainMenuClass;
00017 protected:
00018     virtual void BeginPlay() override;
00019 };
00020
```

## 6.36 Source/CplusplusProject/UI/Controllers/UIController.cpp File Reference

```
#include "UIController.h"
#include "Kismet/GameplayStatics.h"
```

## 6.37 Source/CplusplusProject/UI/Controllers/UIController.h File Reference

```
#include "CoreMinimal.h"
#include "GameFramework/HUD.h"
#include "UIController.generated.h"
```

### Classes

- class [AUIController](#)

## 6.38 UIController.h

[Go to the documentation of this file.](#)

```
00001 #pragma once
00002
00003 #include "CoreMinimal.h"
00004 #include "GameFramework/HUD.h"
00005 #include "UIController.generated.h"
00006
00007 UCLASS(Abstract)
00008 class CPLUSPROJECT_API AUIController : public AHUD
00009 {
00010     GENERATED_BODY()
00011 protected:
00012     // ===== FUNCTIONS =====
00013     FInputModeUIOnly InputModeUI;
00014     FInputModeGameOnly GameInputMode;
00015     // ===== FUNCTIONS =====
00016     void SwitchToUIControls();
00017     void SwitchToPlayerControls();
00018 public:
00019     virtual void CloseAll() {};// tried to do =0 and make the class abstract, getting UIController
    even though I cannot understand where I am instantiating it and I implement the function in the
    inheriting classes
00020
00021 };
```

## 6.39 Source/CplusplusProject/UI/Widgets/GameOverUI.h File Reference

```
#include "CoreMinimal.h"
#include "Blueprint/UserWidget.h"
#include "Components/Button.h"
#include "MainMenuUI.h"
#include "GameOverUI.generated.h"
```

### Classes

- class [UGameOverUI](#)

## 6.40 GameOverUI.h

[Go to the documentation of this file.](#)

```
00001 #pragma once
00002
00003 #include "CoreMinimal.h"
00004 #include "Blueprint/UserWidget.h"
00005 #include "Components/Button.h"
00006 #include "MainMenuUI.h"
00007 #include "GameOverUI.generated.h"
00008
00009 UCLASS()
00010 class CPLUSPROJECT_API UGameOverUI : public UMainMenuUI
00011 {
00012     GENERATED_BODY()
00013
00014     UPROPERTY(VisibleAnywhere, meta = (BindWidget))
00015     UButton* quitGameButton;
00016 private:
00017     // ===== FUNCTIONS =====
00018     UFUNCTION()
00019     inline void QuitGame()
00020     {
00021         GetWorld()->GetFirstPlayerController()->ConsoleCommand("quit");
00022     }
00023     inline virtual void NativeConstruct() override
00024     {
00025         Super::NativeConstruct();
00026         quitGameButton->OnClicked.AddDynamic(this, &UGameOverUI::QuitGame);
00027     }
00028
00029 };
```

## 6.41 Source/CplusplusProject/UI/Widgets/InteractionUI.h File Reference

```
#include "CoreMinimal.h"
#include "Blueprint/UserWidget.h"
#include <Components/TextBlock.h>
#include "InteractionUI.generated.h"
```

### Classes

- class [UInteractionUI](#)

## 6.42 InteractionUI.h

[Go to the documentation of this file.](#)

```
00001 #pragma once
00002
00003 #include "CoreMinimal.h"
00004 #include "Blueprint/UserWidget.h"
00005 #include <Components/TextBlock.h>
00006 #include "InteractionUI.generated.h"
00007
00008 UCLASS()
00009 class CPLUSPROJECT_API UInteractionUI : public UUserWidget
00010 {
00011     GENERATED_BODY()
00012
00013 private:
00014     UPROPERTY(VisibleAnywhere, meta = (BindWidget))
00015     UTextBlock* textBlock;
00016 public:
00017     inline void UpdateWidget(const FText& data) const {textBlock->SetText(data);}
00018
00019 };
```

## 6.43 Source/CplusplusProject/UI/Widgets/LockUI.cpp File Reference

```
#include "LockUI.h"
#include "Kismet/GameplayStatics.h"
#include "../Controllers/GameUIController.h"
#include "CplusplusProject/Environment/DoorOpenComponent.h"
#include "CplusplusProject/Player/MyPlayerController.h"
```

## 6.44 Source/CplusplusProject/UI/Widgets/LockUI.h File Reference

```
#include "CoreMinimal.h"
#include "Blueprint/UserWidget.h"
#include "Components/Button.h"
#include "../Environment/LockControllerComponent.h"
#include "../Environment/AInteractableActor.h"
#include "../Structs/SimonData.h"
#include "Components/AudioComponent.h"
#include "Components/Image.h"
#include "LockUI.generated.h"
```

### Classes

- class [ULockUI](#)

## 6.45 LockUI.h

[Go to the documentation of this file.](#)

```

00001 #pragma once
00002
00003 #include "CoreMinimal.h"
00004 #include "Blueprint/UserWidget.h"
00005 #include "Components/Button.h"
00006 #include "../Environment/LockControllerComponent.h"
00007 #include "../Environment/AInteractableActor.h"
00008 #include "../Structs/SimonData.h"
00009 #include "Components/AudioComponent.h"
00010 #include "Components/Image.h"
00011 #include "LockUI.generated.h"
00012
00013 UCLASS()
00014 class CPLUSPROJECT_API ULockUI : public UUserWidget
00015 {
00016     GENERATED_BODY()
00017 public:
00018     // ===== PROPERTIES =====
00019     UPROPERTY(VisibleAnywhere, meta = (BindWidget), Category = "UI Binding")
00020     UButton* redButton;
00021     UPROPERTY(VisibleAnywhere, meta = (BindWidget), Category = "UI Binding")
00022     UButton* greenButton;
00023     UPROPERTY(VisibleAnywhere, meta = (BindWidget), Category = "UI Binding")
00024     UButton* blueButton;
00025     UPROPERTY(VisibleAnywhere, meta = (BindWidget), Category = "UI Binding")
00026     UButton* yellowButton;
00027     UPROPERTY(VisibleAnywhere, meta = (BindWidget), Category = "UI Binding")
00028     UButton* hintButton;
00029     UPROPERTY(VisibleAnywhere, meta = (BindWidget), Category = "UI Binding")
00030     UButton* closeButton;
00031     UPROPERTY(VisibleAnywhere, meta = (BindWidget), Category = "UI Binding")
00032     UImage* errorImage;
00033     UPROPERTY(VisibleAnywhere, meta = (BindWidget), Category = "UI Binding")
00034     UImage* greenHighlightImage;
00035     UPROPERTY(VisibleAnywhere, meta = (BindWidget), Category = "UI Binding")
00036     UImage* blueHighlightImage;
00037     UPROPERTY(VisibleAnywhere, meta = (BindWidget), Category = "UI Binding")
00038     UImage* yellowHighlightImage;
00039     UPROPERTY(VisibleAnywhere, meta = (BindWidget), Category = "UI Binding")
00040     UImage* redHighlightImage;
00041     UPROPERTY(EditAnywhere, Category = "Audio")
00042     TArray<USoundBase*> notesAudio;
00043     UPROPERTY(EditAnywhere, Category = "Audio")
00044     USoundBase* errorSound;
00045     UPROPERTY(EditAnywhere, Category = "Visuals")
00046     float timeBetweenNotes = 0.7f;
00047     UPROPERTY(EditAnywhere, Category = "Visuals")
00048     float showButtonHighlightTime = 0.7f;
00049     UPROPERTY(EditAnywhere, Category = "Visuals")
00050     float showErrorTime = 1.3f;
00051     UPROPERTY(EditAnywhere, Category = "Audio")
00052     float volumeMultiplier = 1.5;
00053     // ===== PROPERTIES =====
00054     UPROPERTY(EditAnywhere, Category = "Panels")
00055     TSubclassOf<AInteractableActor> openDoorActorClass;
00056     // ===== FUNCTIONS =====
00057     UFUNCTION()
00058     void BindButtons();
00059     UFUNCTION()
00060     void RedButton();
00061     UFUNCTION()
00062     void BlueButton();
00063     UFUNCTION()
00064     void GreenButton();
00065     UFUNCTION()
00066     void YellowButton();
00067     UFUNCTION()
00068     void GiveHint();
00069     UFUNCTION()
00070     void OnOpen();
00071     UFUNCTION()
00072     void OnUnlocked();
00073     UFUNCTION()
00074     void OnClose();
00075     UFUNCTION()
00076     void PlaySequence();
00077     bool CheckSequence();
00078     void SetLockController(ULockControllerComponent* newController);
00079 protected:
00080     // ===== FUNCTIONS =====
00081     virtual void NativeConstruct() override;
00082 private:

```



```

00087 // ===== PROPERTIES =====
00088 UPROPERTY()
00090 ULockControllerComponent* lockController = nullptr;
00091 UPROPERTY()
00092 UAudioComponent* audioComponent= nullptr;
00093 UPROPERTY()
00095 TArray<Notes> sequence;
00096 UPROPERTY()
00097 class AGameUIController* uiController; // circular dependencies
00098 FTimerHandle imageTimerHandle;
00099 FTimerHandle playTimerHandle;
00100 FTimerDelegate playNoteDelegate;
00101 // ===== FUNCTIONS =====
00102 bool ValidityChecks(const ULockControllerComponent* newController) const;
00103 void PlayNote(int8 noteNumber);
00104 void ResetSequence();
00105 void AddToSequence(Notes note);
00106 void ShowWrongSequenceUI();
00107 void OnButtonClick(ButtonColors color);
00108 void DebugSequences();
00109 void ShowImage( UImage* image, bool hideAutomatically = true, float delay = 0.5f);
00110 void HideImage( UImage* image);
00111 void PlayNextSound(int i);
00112 void ShowHighlightEffect(ButtonColors color);
00113 void HideHightlightEffect();
00114 ButtonColors FindColorByNote(Notes note) const;
00115 };

```

## 6.46 Source/CplusplusProject/UI/Widgets/MainMenuUI.cpp File Reference

```

#include "MainMenuUI.h"
#include "Kismet/GameplayStatics.h"

```

## 6.47 Source/CplusplusProject/UI/Widgets/MainMenuUI.h File Reference

```

#include "CoreMinimal.h"
#include "Blueprint/UserWidget.h"
#include "Components/Button.h"
#include "MainMenuUI.generated.h"

```

### Classes

- class [UMainMenuUI](#)

## 6.48 MainMenuUI.h

[Go to the documentation of this file.](#)

```

00001 #pragma once
00002
00003 #include "CoreMinimal.h"
00004 #include "Blueprint/UserWidget.h"
00005 #include "Components/Button.h"
00006 #include "MainMenuUI.generated.h"
00007
00008 UCLASS()
00009 class CPLUSPROJECT_API UMainMenuUI : public UUserWidget
00010 {
00011     GENERATED_BODY()
00012 public:
00013 // ===== PROPERTIES =====

```

```
00014     UPROPERTY(VisibleAnywhere, meta = (BindWidget))
00015     UButton* startGameButton;
00016     UFUNCTION()
00017     void StartGame();
00018 protected:
00019     // ===== FUNCTIONS =====
00020     virtual void NativeConstruct() override;
00021
00022
00023 };
```

# Index

## A

- SimonData.h, [41](#)
- AAInteractableActor, [9](#)
  - AAInteractableActor, [9](#)
  - BeginFocus2D, [10](#)
  - EndFocus2D, [10](#)
  - GetItemData, [10](#)
  - itemData, [10](#)
- Ab
  - SimonData.h, [41](#)
- actionInteract
  - AMyPlayerController, [18](#)
- actionMove
  - AMyPlayerController, [18](#)
- AFlickeringLight, [10](#)
  - AFlickeringLight, [11](#)
  - BeginPlay, [11](#)
  - spotlight, [11](#)
  - Tick, [11](#)
- AGameOverUIController, [12](#)
  - BeginPlay, [12](#)
  - gameOverUIClass, [13](#)
- AGameUIContoller, [13](#)
  - BeginPlay, [14](#)
  - CloseAll, [14](#)
  - CloseEncourageInteractUI, [14](#)
  - CloseLockUI, [14](#)
  - enourageInteractionUIClass, [15](#)
  - interactionUIClass, [15](#)
  - lockUIClass, [15](#)
  - OpenEncourageInteractUI, [14](#)
  - SetLockController, [14](#)
  - UpdateInteractionUI, [14](#)
- AMainMenuUIController, [15](#)
  - BeginPlay, [16](#)
  - mainMenuClass, [16](#)
- AMyPlayerController, [16](#)
  - actionInteract, [18](#)
  - actionMove, [18](#)
  - AMyPlayerController, [17](#)
  - direction, [18](#)
  - inputMappingContext, [18](#)
  - OnPossess, [17](#)
  - OnUnPossess, [17](#)
  - SetLastInteractedActor, [17](#)
- AUIController, [18](#)
  - CloseAll, [19](#)
  - gameInputMode, [19](#)
  - inputModeUI, [19](#)

- SwitchToPlayerControls, [19](#)
- SwitchToUIControls, [19](#)

## B

- SimonData.h, [41](#)
- Bb
  - SimonData.h, [41](#)
- BeginFocus2D
  - AAInteractableActor, [10](#)
  - IInteractionInterface, [23](#)
- BeginPlay
  - AFlickeringLight, [11](#)
  - AGameOverUIController, [12](#)
  - AGameUIContoller, [14](#)
  - AMainMenuUIController, [16](#)
- BindButtons
  - ULockUI, [26](#)
- Blue
  - SimonData.h, [41](#)
- BlueButton
  - ULockUI, [26](#)
- blueButton
  - ULockUI, [28](#)
- blueHighlightImage
  - ULockUI, [28](#)
- ButtonColors
  - SimonData.h, [40](#)

## C

- SimonData.h, [41](#)
- CheckSequence
  - ULockUI, [27](#)
- CloseAll
  - AGameUIContoller, [14](#)
  - AUIController, [19](#)
- closeButton
  - ULockUI, [28](#)
- CloseEncourageInteractUI
  - AGameUIContoller, [14](#)
- CloseLockUI
  - AGameUIContoller, [14](#)
- CPlusProject, [1](#)
- CplusProject, [19](#)
  - CplusProject, [20](#)
- CplusProject.cpp
  - IMPLEMENT\_PRIMARY\_GAME\_MODULE, [33](#)

## D

- SimonData.h, [41](#)
- Db

- SimonData.h, [41](#)
- description
  - FItemData, [21](#)
- direction
  - AMyPlayerController, [18](#)
- E
  - SimonData.h, [41](#)
- Eb
  - SimonData.h, [41](#)
- EndFocus2D
  - AAInteractableActor, [10](#)
  - IInteractionInterface, [23](#)
- enourageInteractionUIClass
  - AGameUIContoller, [15](#)
- errorImage
  - ULockUI, [28](#)
- errorSound
  - ULockUI, [28](#)
- F
  - SimonData.h, [41](#)
- FItemData, [20](#)
  - description, [21](#)
  - FItemData, [20](#)
  - GENERATED\_BODY, [21](#)
  - interactable, [21](#)
  - itemType, [21](#)
- FSimonData, [21](#)
  - FSimonData, [22](#)
  - GENERATED\_BODY, [22](#)
  - mapColorToNote, [22](#)
  - sequence, [22](#)
- G
  - SimonData.h, [41](#)
- gameInputMode
  - AUIController, [19](#)
- gameOverUIClass
  - AGameOverUIController, [13](#)
- Gb
  - SimonData.h, [41](#)
- GENERATED\_BODY
  - FItemData, [21](#)
  - FSimonData, [22](#)
- GetItemData
  - AAInteractableActor, [10](#)
  - IInteractionInterface, [23](#)
- GiveHint
  - ULockUI, [27](#)
- Green
  - SimonData.h, [41](#)
- GreenButton
  - ULockUI, [27](#)
- greenButton
  - ULockUI, [28](#)
- greenHighlightImage
  - ULockUI, [28](#)
- hintButton
  - ULockUI, [29](#)
- IInteractionInterface, [22](#)
  - BeginFocus2D, [23](#)
  - EndFocus2D, [23](#)
  - GetItemData, [23](#)
- IMPLEMENT\_PRIMARY\_GAME\_MODULE
  - CplusplusProject.cpp, [33](#)
- inputMappingContext
  - AMyPlayerController, [18](#)
- inputModeUI
  - AUIController, [19](#)
- Interactable
  - ItemData.h, [39](#)
- interactable
  - FItemData, [21](#)
- interactionUIClass
  - AGameUIContoller, [15](#)
- itemData
  - AAInteractableActor, [10](#)
- ItemData.h
  - Interactable, [39](#)
  - ItemType, [39](#)
  - Lock, [39](#)
  - NPC, [39](#)
  - OpenDoor, [39](#)
  - Pickup, [39](#)
  - Static, [39](#)
- ItemType
  - ItemData.h, [39](#)
- itemType
  - FItemData, [21](#)
- Lock
  - ItemData.h, [39](#)
- lockUIClass
  - AGameUIContoller, [15](#)
- mainMenuClass
  - AMainMenuUIController, [16](#)
- mapColorToNote
  - FSimonData, [22](#)
- NativeConstruct
  - ULockUI, [27](#)
  - UMainMenuUI, [30](#)
- None
  - SimonData.h, [41](#)
- Notes
  - SimonData.h, [41](#)
- notesAudio
  - ULockUI, [29](#)
- NPC
  - ItemData.h, [39](#)
- OnClose
  - ULockUI, [27](#)
- OnOpen

- ULockUI, 27
- OnPossess
  - AMyPlayerController, 17
- OnUnlocked
  - ULockUI, 27
- OnUnPossess
  - AMyPlayerController, 17
- OpenDoor
  - ItemData.h, 39
- openDoorActorClass
  - ULockUI, 29
- OpenEncourageInteractUI
  - AGameUIContoller, 14
- Pickup
  - ItemData.h, 39
- PlaySequence
  - ULockUI, 27
- README.txt, 33
- Red
  - SimonData.h, 41
- RedButton
  - ULockUI, 28
- redButton
  - ULockUI, 29
- redHighlightImage
  - ULockUI, 29
- sequence
  - FSimonData, 22
- SetLastInteractedActor
  - AMyPlayerController, 17
- SetLockController
  - AGameUIContoller, 14
  - ULockUI, 28
- showButtonHightlightTime
  - ULockUI, 29
- showErrorTime
  - ULockUI, 29
- SimonData.h
  - A, 41
  - Ab, 41
  - B, 41
  - Bb, 41
  - Blue, 41
  - ButtonColors, 40
  - C, 41
  - D, 41
  - Db, 41
  - E, 41
  - Eb, 41
  - F, 41
  - G, 41
  - Gb, 41
  - Green, 41
  - None, 41
  - Notes, 41
  - Red, 41
  - Yellow, 41
- Source/CplusplusProject/CplusplusProject.Build.cs, 33
- Source/CplusplusProject/CplusplusProject.cpp, 33
- Source/CplusplusProject/CplusplusProject.h, 33, 34
- Source/CplusplusProject/Environment/AInteractableActor.cpp, 34
- Source/CplusplusProject/Environment/AInteractableActor.h, 34
- Source/CplusplusProject/Environment/DoorOpenComponent.h, 35
- Source/CplusplusProject/Environment/FlickeringLight.cpp, 35
- Source/CplusplusProject/Environment/FlickeringLight.h, 35, 36
- Source/CplusplusProject/Environment/LockControllerComponent.h, 36
- Source/CplusplusProject/Interfaces/InteractionInterface.h, 37
- Source/CplusplusProject/Player/MyPlayerController.cpp, 37
- Source/CplusplusProject/Player/MyPlayerController.h, 38
- Source/CplusplusProject/Structs/ItemData.cpp, 39
- Source/CplusplusProject/Structs/ItemData.h, 39, 40
- Source/CplusplusProject/Structs/SimonData.cpp, 40
- Source/CplusplusProject/Structs/SimonData.h, 40, 41
- Source/CplusplusProject/UI/Controllers/GameOverUIController.cpp, 42
- Source/CplusplusProject/UI/Controllers/GameOverUIController.h, 42
- Source/CplusplusProject/UI/Controllers/GameUIContoller.cpp, 43
- Source/CplusplusProject/UI/Controllers/GameUIContoller.h, 43
- Source/CplusplusProject/UI/Controllers/MainMenuUIController.cpp, 44
- Source/CplusplusProject/UI/Controllers/MainMenuUIController.h, 44, 45
- Source/CplusplusProject/UI/Controllers/UIController.cpp, 45
- Source/CplusplusProject/UI/Controllers/UIController.h, 45
- Source/CplusplusProject/UI/Widgets/GameOverUI.h, 46
- Source/CplusplusProject/UI/Widgets/InteractionUI.h, 46, 47
- Source/CplusplusProject/UI/Widgets/LockUI.cpp, 47
- Source/CplusplusProject/UI/Widgets/LockUI.h, 47, 48
- Source/CplusplusProject/UI/Widgets/MainMenuUI.cpp, 49
- Source/CplusplusProject/UI/Widgets/MainMenuUI.h, 49
- spotlight
  - AFlickeringLight, 11
- StartGame
  - UMainMenuUI, 30
- startGameButton
  - UMainMenuUI, 31
- Static
  - ItemData.h, 39
- SwitchToPlayerControls
  - AUIController, 19
- SwitchToUIControls
  - AUIController, 19
- Tick
  - AFlickeringLight, 11

- timeBetweenNotes
  - ULockUI, [29](#)
- UGameOverUI, [23](#)
- UInteractionInterface, [24](#)
- UInteractionUI, [25](#)
  - UpdateWidget, [25](#)
- ULockUI, [25](#)
  - BindButtons, [26](#)
  - BlueButton, [26](#)
  - blueButton, [28](#)
  - blueHighlightImage, [28](#)
  - CheckSequence, [27](#)
  - closeButton, [28](#)
  - errorImage, [28](#)
  - errorSound, [28](#)
  - GiveHint, [27](#)
  - GreenButton, [27](#)
  - greenButton, [28](#)
  - greenHighlightImage, [28](#)
  - hintButton, [29](#)
  - NativeConstruct, [27](#)
  - notesAudio, [29](#)
  - OnClose, [27](#)
  - OnOpen, [27](#)
  - OnUnlocked, [27](#)
  - openDoorActorClass, [29](#)
  - PlaySequence, [27](#)
  - RedButton, [28](#)
  - redButton, [29](#)
  - redHighlightImage, [29](#)
  - SetLockController, [28](#)
  - showButtonHightlightTime, [29](#)
  - showErrorTime, [29](#)
  - timeBetweenNotes, [29](#)
  - volumeMultiplier, [29](#)
  - YellowButton, [28](#)
  - yellowButton, [29](#)
  - yellowHighlightImage, [30](#)
- UMainMenuUI, [30](#)
  - NativeConstruct, [30](#)
  - StartGame, [30](#)
  - startGameButton, [31](#)
- UpdateInteractionUI
  - AGameUIContoller, [14](#)
- UpdateWidget
  - UInteractionUI, [25](#)
- volumeMultiplier
  - ULockUI, [29](#)
- Yellow
  - SimonData.h, [41](#)
- YellowButton
  - ULockUI, [28](#)
- yellowButton
  - ULockUI, [29](#)
- yellowHighlightImage
  - ULockUI, [30](#)